QBS 103 Final (clean)

2025-08-18

```
library(readr)
proj_genes <- read_csv("/Users/ayei/Desktop/QBS103_GSE157103_genes.csv")
```

```
## New names:
## Rows: 100 Columns: 127
## -- Column specification
## -------------------------------------------------------- Delimiter: "," chr
## (1): ...1 dbl (126): COVID_01_39y_male_NonICU, COVID_02_63y_male_NonICU,
## COVID_03_33y_...
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * `` -> `...1`
```

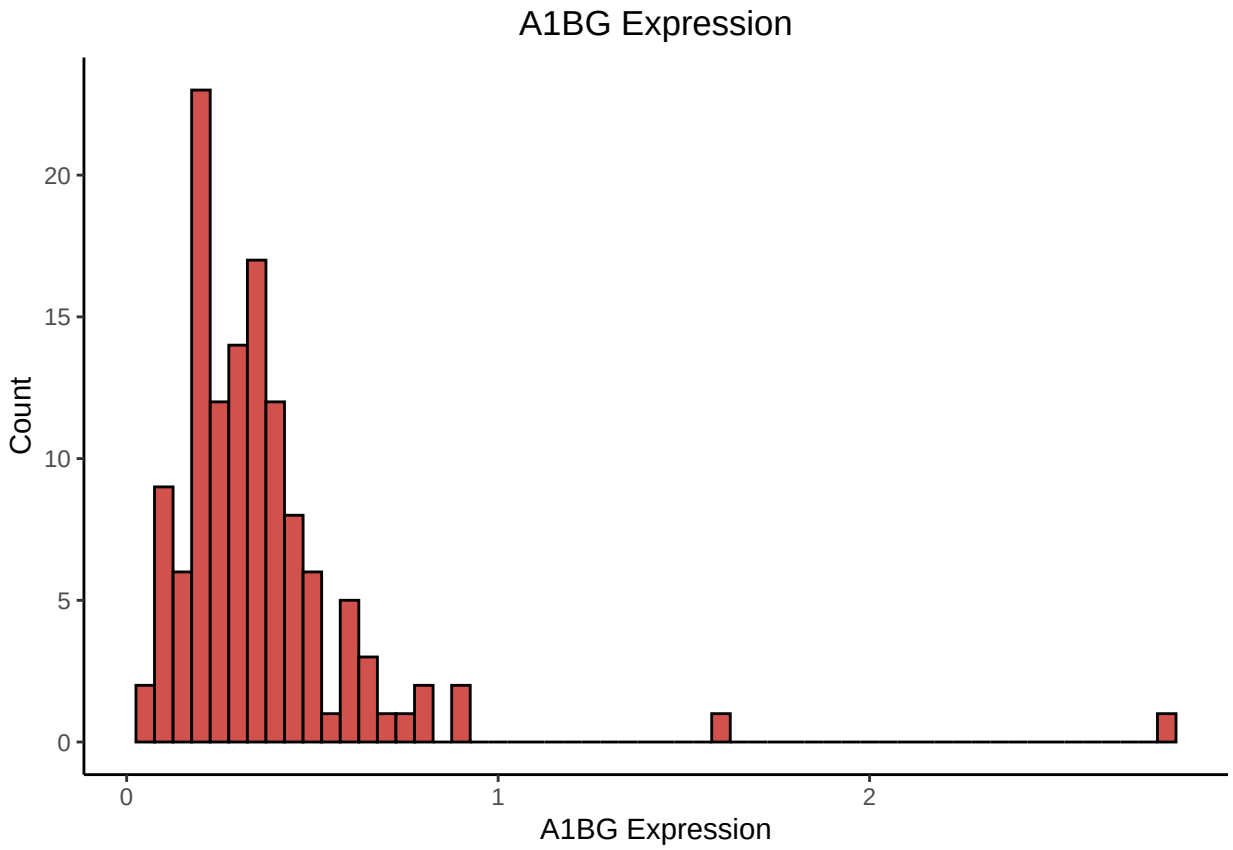
```
library(readr)
proj_matrix <- read_csv("/Users/ayei/Desktop/QBS103_GSE157103_series_matrix-1.csv")
```

```
## Rows: 126 Columns: 25
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (21): participant_id, geo_accession, status, !Sample_submission_date, la...
## dbl  (4): channel_count, charlson_score, ventilator-free_days, hospital-free...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
# Because I need to match the participant_id column/row position in order to link the two data sets, I

# Converting raw data to data frame
proj_genes_df <- as.data.frame(proj_genes)

# Setting the row names as the gene names
rownames(proj_genes_df) <- proj_genes_df[[1]]

# Now I have to remove the first column since I am using them as row names
proj_genes_df <- proj_genes_df[ , -1]

# Transposing the data so that the gene names go across the top and the participant ID's became the col
new_proj_genes <- as.data.frame(t(proj_genes_df))

# Just doudble checking
# colnames(proj_matrix)
# colnames(new_proj_genes)

# Now I am adding a new column for "participant_id" to "new_proj_genes" with the row names, so that whe
new_proj_genes$participant_id <- rownames(new_proj_genes)
```

```r
# Merging the two datasets
merged_proj_data <- merge(proj_matrix, new_proj_genes, by = "participant_id")

# Double checking to see if the merge was successful
# summary(merged_proj_data)
```

```r
library(ggplot2)

# Making histogram for A1BG expression
ggplot(merged_proj_data, aes(x = A1BG)) +
  # Decided to make my binwidth 0.05 for more data points/make the data points more distinct
  # Also customized my colors
  geom_histogram(binwidth = 0.05, fill = "#D1514D", color = "black") +
  # Added a title, and x and y axis labels
  labs(title = "A1BG Expression",
       x = "A1BG Expression", y = "Count") +
  # Set the theme to classic for a clean background
  theme_classic() +
  # Wanted to center my title
     ##https://www.r-bloggers.com/2025/03/how-to-center-ggplot-title-subtitle-and-caption-in-ggplot2-wi
  theme(plot.title = element_text(hjust = 0.5))
```

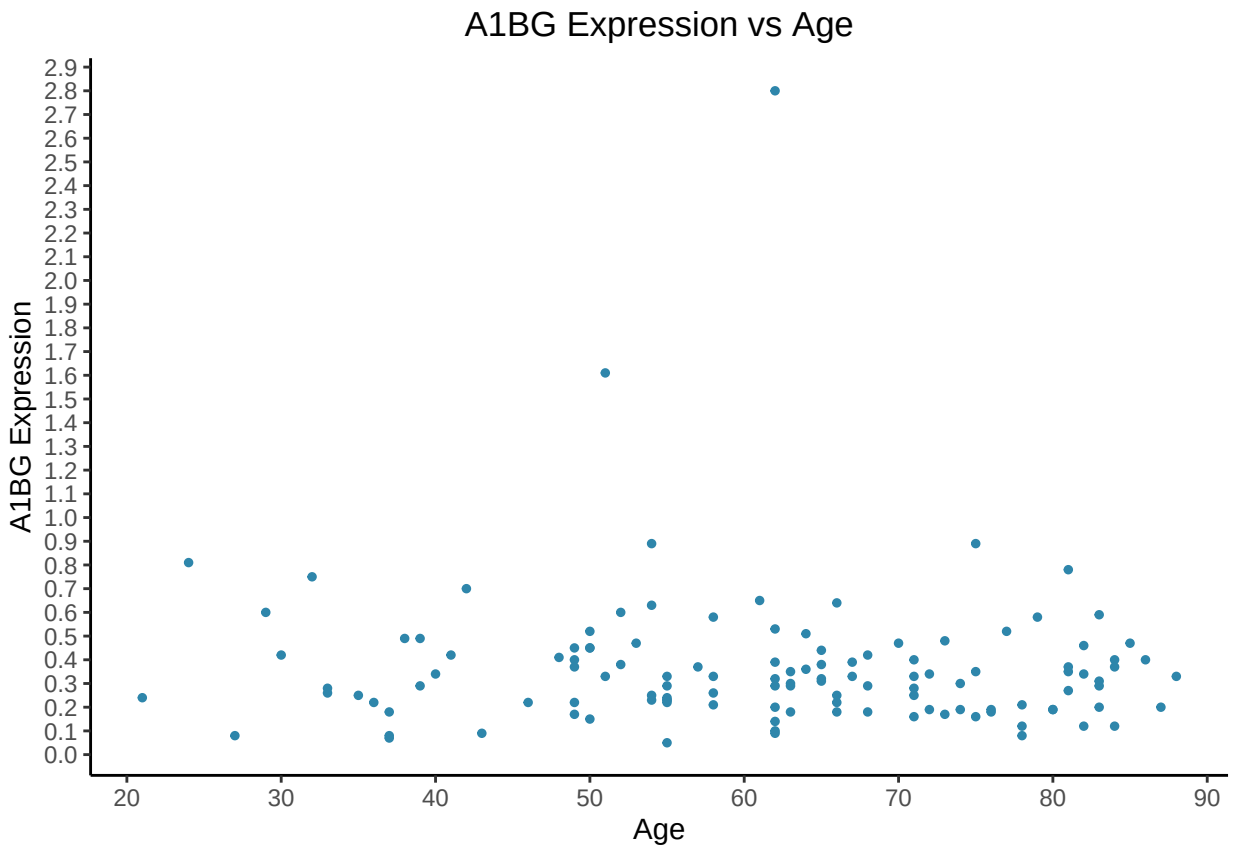



```r
# "age" was not being registered as a number, so I had to change the age column to numeric
merged_proj_data$age <- as.numeric(merged_proj_data$age)
```

```
## Warning: NAs introduced by coercion
```

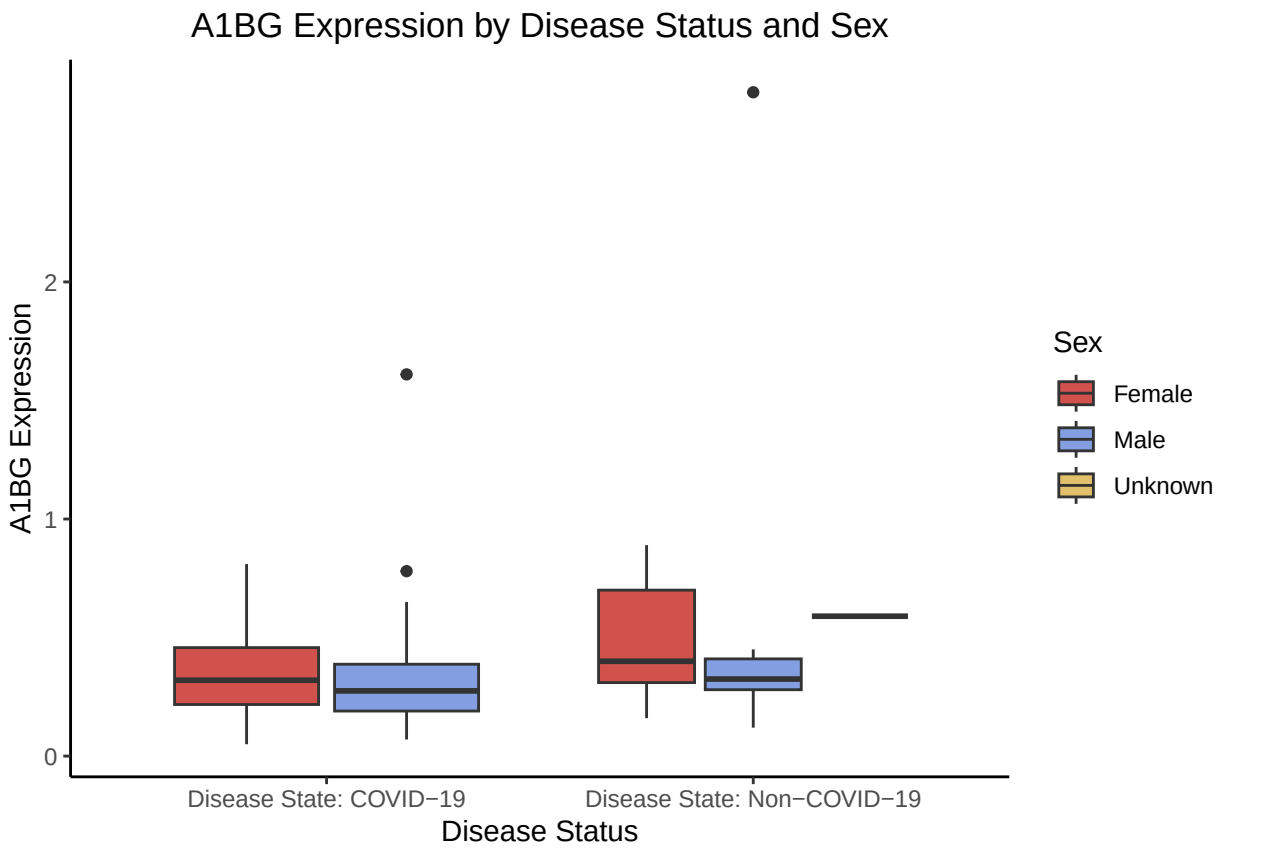
```r
# Making scatterplot for A1BG Expression vs Age
ggplot(merged_proj_data, aes(x = age, y = A1BG)) +
  # Customized the color and the size -- I set the size to 1 so that the point is more precise (when it
  geom_point(color = "#2E86AB", size = 1) +
  # Added a title, and x and y axis labels
  labs(title = "A1BG Expression vs Age",
       x = "Age", y = "A1BG Expression") +
  # Customized the scale for x axis to go up by 10s to make the plot easier to read
      ##https://www.sthda.com/english/wiki/ggplot2-axis-scales-and-transformations#google_vignette
  scale_x_continuous(breaks = seq(0, 100, by = 10)) +
  # Customized the scale for y axis to go up by 0.1 to make the plot easier to read
  scale_y_continuous(breaks = seq(0, 3, by = 0.1)) +
  # Set my theme to classic for a clean background
  theme_classic() +
  # Centered the title
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Removed 3 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



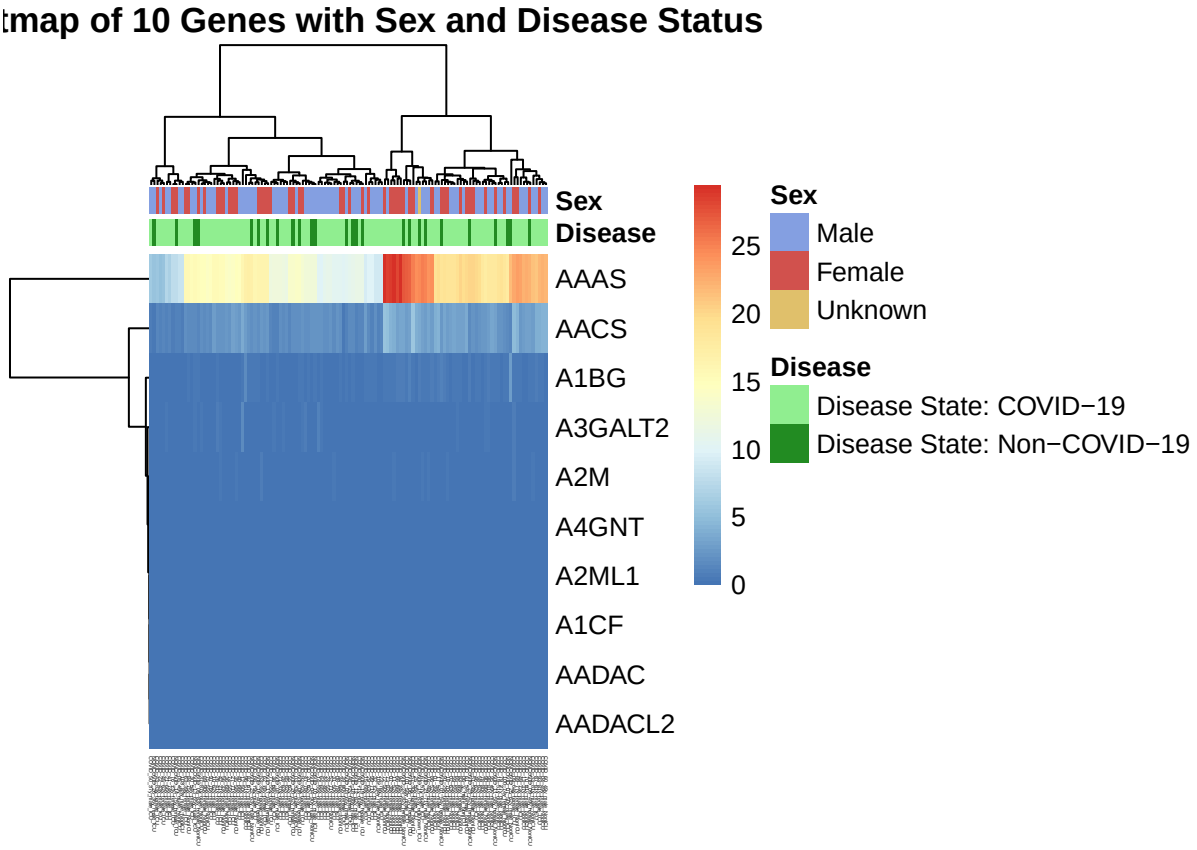

```r
# Making boxplot for A1BG expression by disease status and sex
ggplot(merged_proj_data, aes(x = disease_status, y = A1BG, fill = sex)) +
  geom_boxplot() +
  # Customized the colors for sex
  scale_fill_manual(values = c("male"= '#849FE1', "female" = '#D1514D',"unknown" = '#E0C06B'),
  labels = function(x) tools::toTitleCase(as.character(x))) +
  scale_x_discrete(labels = function(x) tools::toTitleCase(as.character(x))) +
```

```
  # Added a title, and x and y labels
  labs(title = "A1BG Expression by Disease Status and Sex",
       x = "Disease Status", y = "A1BG Expression", fill = "Sex") +
  # Set my theme to classic for a clean background
  theme_classic() +
  # Centered my title
  theme(plot.title = element_text(hjust = 0.5))
```




```
# install.packages('pheatmap')
library(pheatmap)

# Specifying ten genes for heatmap
ten_genes <- c("A1BG", "A1CF", "A2M", "A2ML1", "A3GALT2", "A4GNT", "AAAS", "AACS", "AADAC", "AADACL2")

# Getting expression values for the ten genes
gene_value_matrix <- merged_proj_data[, ten_genes]

# Making the participant ID's as the rownames
rownames(gene_value_matrix) <- merged_proj_data$participant_id

# Making mini data frame to match the participants w their disease status and sex
annotationData <- data.frame(
  row.names = merged_proj_data$participant_id,
  Disease = merged_proj_data$disease_status,
  Sex = merged_proj_data$sex)

annotationData$Disease <- as.character(annotationData$Disease)
```

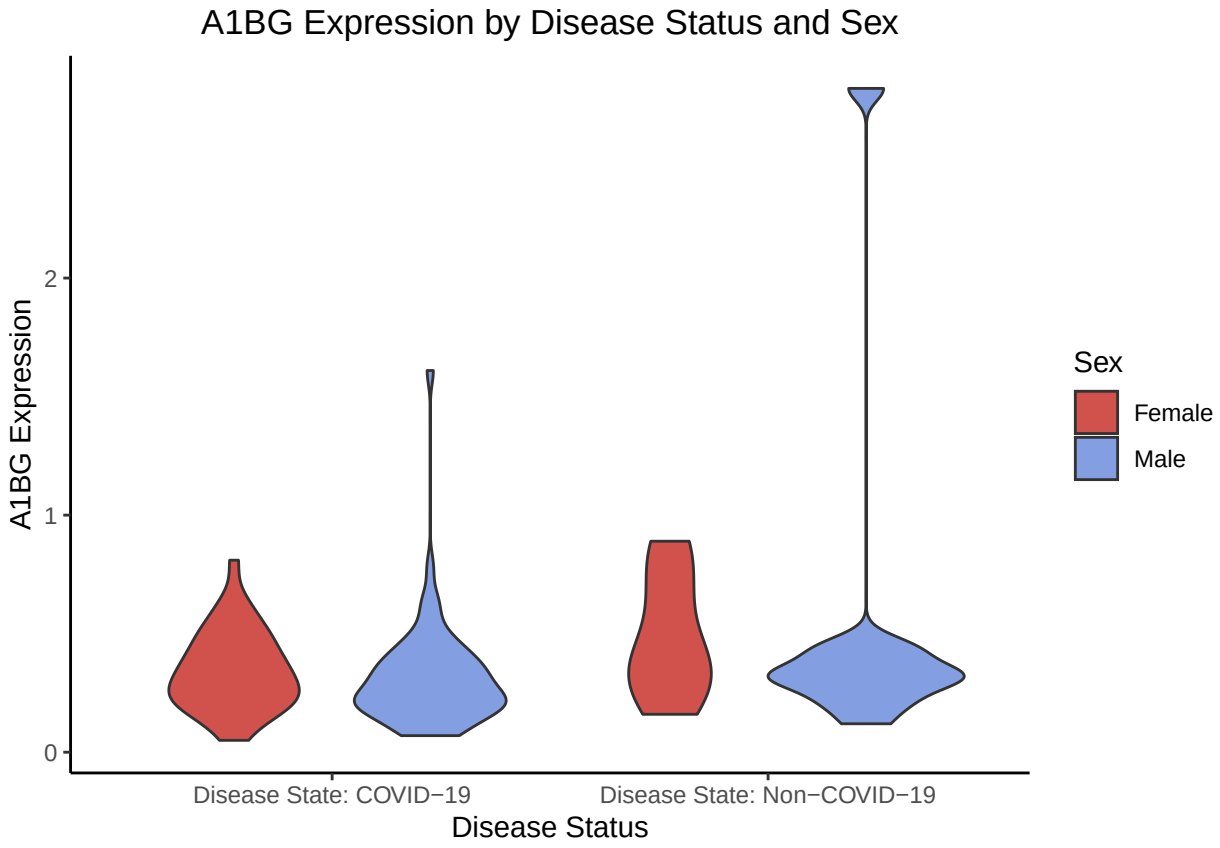
```r
annotationData$Disease <- factor(annotationData$Disease,
  levels = c("disease state: COVID-19", "disease state: non-COVID-19"),
  labels = tools::toTitleCase(c("disease state: COVID-19", "disease state: non-COVID-19")))

annotationData$Sex <- as.character(annotationData$Sex)
annotationData$Sex <- factor(annotationData$Sex,
  levels = c("male", "female", "unknown"),
  labels = tools::toTitleCase(c("male", "female", "unknown")))

# Setting colors for each of the variables
annotationColors <- list(
  Disease = c("Disease State: COVID-19" = "lightgreen", "Disease State: Non-COVID-19" = "forestgreen"),
  Sex = c("Male"= '#849FE1', "Female" = '#D1514D',"Unknown" = '#E0C06B'))

# Making the heatmap
pheatmap(
  t(gene_value_matrix),
  annotation_col = annotationData,
  annotation_colors = annotationColors,
  fontsize_col = 2,
  main = "Heatmap of 10 Genes with Sex and Disease Status")
```




```r
library(ggplot2)

# Making violin plot for A1BG expression by disease status and sex
ggplot(merged_proj_data, aes(x = disease_status, y = A1BG, fill = sex)) +
  geom_violin()+
```

```
  # Customized the colors for sex
  scale_fill_manual(values = c("male"= '#849FE1', "female" = '#D1514D',"unknown" = '#E0C06B'),
  labels = function(x) tools::toTitleCase(as.character(x))) +
  scale_x_discrete(labels = function(x) tools::toTitleCase(as.character(x))) +
  # Added a title, and x and y labels
  labs(title = "A1BG Expression by Disease Status and Sex",
       x = "Disease Status", y = "A1BG Expression", fill = "Sex") +
  # Set my theme to classic for a clean background
  theme_classic() +
  # Centered my title
  theme(plot.title = element_text(hjust = 0.5))
```

```
## Warning: Groups with fewer than two datapoints have been dropped.
## i Set `drop = FALSE` to consider such groups for position adjustment purposes.
```




```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(knitr)
library(kableExtra)
```

```
##
## Attaching package: 'kableExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     group_rows
```

```r
set.seed(103)

# Making the summary table
my_table <- merged_proj_data %>%
  # Capitalizing the icu status
  mutate(icu_status = gsub("_", " ", icu_status),
         icu_status = tools::toTitleCase(tolower(icu_status))) %>%
  # Stratifying using icu status
  group_by(icu_status) %>%
  summarise(
    n = n(),

    # 1st categorical variable
    male.n = sum(sex == "male"),
    male.pct = round(100*mean(sex == "male"), 1),

    # 2nd categorical variable
    disease_status.yes = sum(disease_status == "disease state: COVID-19"),
    disease_status.yes.pct = round(100*mean(disease_status == "disease state: COVID-19"), 1),

    # 1st continuous variable
    age.mean = round(mean(age, na.rm = TRUE), 1),
    age.sd = round(sd(age, na.rm = TRUE), 1),

    # 2nd continuous variable
    gene.mean = round(mean(A1BG, na.rm = TRUE), 2),    # changed here
    gene.sd = round(sd(A1BG, na.rm = TRUE), 2),

    # 3rd continuous variable
    charlson_score.mean = round(mean(charlson_score, na.rm = TRUE), 2), # changed here
    charlson_score.sd = round(sd(charlson_score, na.rm = TRUE), 2)
  ) %>%

  mutate(
    'Sex (male)' = paste0(male.n, " (", male.pct, "%)"),
    'Disease status (COVID-19)' = paste0(disease_status.yes, " (", disease_status.yes.pct, "%)"),
    'Age' = paste0(age.mean, " (", age.sd, ")"),
    'A1BG Gene Expression' = paste0(gene.mean, " (", gene.sd, ")"),
    'Charlson Score' = paste0(charlson_score.mean, " (", charlson_score.sd, ")")
  ) %>%

  select(icu_status, n, 'Sex (male)', 'Disease status (COVID-19)', 'Age', 'A1BG Gene Expression', 'Charl
```

```r
# Making LaTex table
print_table <- my_table
```

```r
# Editing the column name for icu status
colnames(print_table)[colnames(print_table) == "icu_status"] <- "ICU Status"

kable(print_table, format = "latex", booktabs = TRUE,
      caption = "Summary Statistics Stratified by ICU Status",
      col.names = colnames(print_table)) %>%
  kable_styling(latex_options = c("hold_position", "striped"))
```

Table 1: Summary Statistics Stratified by ICU Status

ICU Status	n	Sex (male)	Disease status (COVID-19)	Age	A1BG Gene Expression	Charlson Score
No	60	33 (55%)	50 (83.3%)	58.7 (17.8)	0.44 (0.39)	3.15 (2.46)
Yes	66	41 (62.1%)	50 (75.8%)	63.5 (14)	0.29 (0.16)	3.82 (2.5)

```r
citation("readr")
```

```
## To cite package 'readr' in publications use:
##
##   Wickham H, Hester J, Bryan J (2024). _readr: Read Rectangular Text
##   Data_. doi:10.32614/CRAN.package.readr
##   <https://doi.org/10.32614/CRAN.package.readr>, R package version
##   2.1.5, <https://CRAN.R-project.org/package=readr>.
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {readr: Read Rectangular Text Data},
##     author = {Hadley Wickham and Jim Hester and Jennifer Bryan},
##     year = {2024},
##     note = {R package version 2.1.5},
##     url = {https://CRAN.R-project.org/package=readr},
##     doi = {10.32614/CRAN.package.readr},
##   }
```

```r
citation("ggplot2")
```

```
## To cite ggplot2 in publications, please use
##
##   H. Wickham. ggplot2: Elegant Graphics for Data Analysis.
##   Springer-Verlag New York, 2016.
##
## A BibTeX entry for LaTeX users is
##
##   @Book{,
##     author = {Hadley Wickham},
##     title = {ggplot2: Elegant Graphics for Data Analysis},
##     publisher = {Springer-Verlag New York},
##     year = {2016},
##     isbn = {978-3-319-24277-4},
##     url = {https://ggplot2.tidyverse.org},
##   }
```

```r
citation("pheatmap")
```

```
## To cite package 'pheatmap' in publications use:
##
##   Kolde R (2025). _pheatmap: Pretty Heatmaps_.
##   doi:10.32614/CRAN.package.pheatmap
##   <https://doi.org/10.32614/CRAN.package.pheatmap>, R package version
##   1.0.13, <https://CRAN.R-project.org/package=pheatmap>.
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {pheatmap: Pretty Heatmaps},
##     author = {Raivo Kolde},
##     year = {2025},
##     note = {R package version 1.0.13},
##     url = {https://CRAN.R-project.org/package=pheatmap},
##     doi = {10.32614/CRAN.package.pheatmap},
##   }
```

```r
citation("knitr")
```

```
## To cite package 'knitr' in publications use:
##
##   Xie Y (2025). _knitr: A General-Purpose Package for Dynamic Report
##   Generation in R_. R package version 1.50, <https://yihui.org/knitr/>.
##
##   Yihui Xie (2015) Dynamic Documents with R and knitr. 2nd edition.
##   Chapman and Hall/CRC. ISBN 978-1498716963
##
##   Yihui Xie (2014) knitr: A Comprehensive Tool for Reproducible
##   Research in R. In Victoria Stodden, Friedrich Leisch and Roger D.
##   Peng, editors, Implementing Reproducible Computational Research.
##   Chapman and Hall/CRC. ISBN 978-1466561595
##
## To see these entries in BibTeX format, use 'print(<citation>,
## bibtex=TRUE)', 'toBibtex(.)', or set
## 'options(citation.bibtex.max=999)'.
```

```r
citation("dplyr")
```

```
## To cite package 'dplyr' in publications use:
##
##   Wickham H, François R, Henry L, Müller K, Vaughan D (2023). _dplyr: A
##   Grammar of Data Manipulation_. doi:10.32614/CRAN.package.dplyr
##   <https://doi.org/10.32614/CRAN.package.dplyr>, R package version
##   1.1.4, <https://CRAN.R-project.org/package=dplyr>.
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {dplyr: A Grammar of Data Manipulation},
##     author = {Hadley Wickham and Romain François and Lionel Henry and Kirill Müller and Davis Vaughan
##     year = {2023},
##     note = {R package version 1.1.4},
```

```
##     url = {https://CRAN.R-project.org/package=dplyr},
##     doi = {10.32614/CRAN.package.dplyr},
##   }
```

```
citation("kableExtra")
```

```
## To cite package 'kableExtra' in publications use:
##
##   Zhu H (2024). _kableExtra: Construct Complex Table with 'kable' and
##   Pipe Syntax_. doi:10.32614/CRAN.package.kableExtra
##   <https://doi.org/10.32614/CRAN.package.kableExtra>, R package version
##   1.4.0, <https://CRAN.R-project.org/package=kableExtra>.
##
## A BibTeX entry for LaTeX users is
##
##   @Manual{,
##     title = {kableExtra: Construct Complex Table with 'kable' and Pipe Syntax},
##     author = {Hao Zhu},
##     year = {2024},
##     note = {R package version 1.4.0},
##     url = {https://CRAN.R-project.org/package=kableExtra},
##     doi = {10.32614/CRAN.package.kableExtra},
##   }
```